



**Embedded Linux  
Conference**  
North America

# Creating BT PAN/RNDIS router using OpenWrt

**Koichi Okamoto, Ishikawa Masayuki /  
let's dive into their mechanism!**

# Biography (Koichi Okamoto)

- Architect of network team (IP layer or higher) of our Video & Sound platform based on Linux at Sony Home Entertainment and Sound Products Inc.



See the below URL for more detailed with Sony products related to me

<https://ossna19.sched.com/speaker/koichiokamoto>

# Table of Contents (Two parts)

- Off-the-shelf hardware using OpenWrt extends BT PAN in addition to USB RNDIS
  - What is Bluetooth (BT) PAN
  - Learning Linux BT mechanisms
  - Overview
  - Hardware/Software Requirement
  - Create OpenWrt Firmware
  - Write OpenWrt firmware
  - *bluetoothctl* utility flexibility
  - How to control BT PAN on BlueZ
  - Persistence of Pairing Information
  - Make NAP work at cold start
- Introduction to low-end devices running NuttX with OpenWrt router
  - Introduction to NuttX networking features
  - How to run Bluetooth on NuttX
  - Software stack for Bluetooth networking
  - BTstack log example
  - MP3 streaming via Bluetooth
  - Running RNDIS on NuttX
  - Working with Wi-Fi on Spresense
  - Use case for Webserver via Wi-Fi

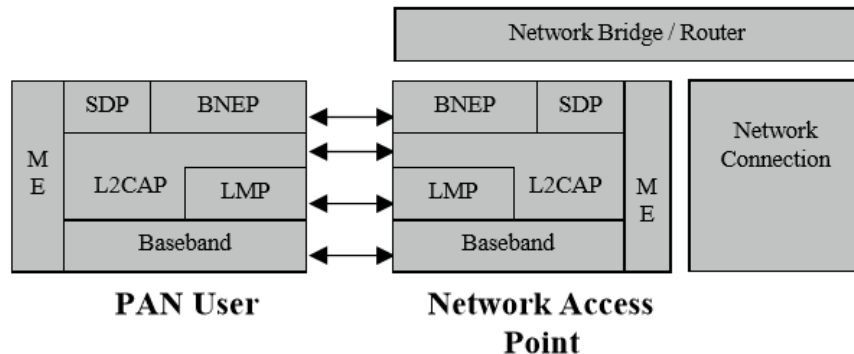
# Off-the-self hardware using OpenWrt extends BT PAN in addition to USB RNDIS

# What is Bluetooth (BT) PAN

- PAN stands for Personal Area Networking

[https://www.bluetooth.org/docman/handlers/DownloadDoc.ashx?doc\\_id=6554](https://www.bluetooth.org/docman/handlers/DownloadDoc.ashx?doc_id=6554)

- The following roles are defined within the PAN profile.
  - Network Access Point (NAP) and NAP service
  - Group Ad-hoc Network (GN) and GN service
  - PAN User (PANU) and PANU service



# Learning Linux BT mechanisms (1/2)

Dell OptiPlex7040  
Core i7-6700 3.4GHz  
RAM 8GByte  
HDD 1TByte

Ubuntu16.04.5 LTS (xenial)



BT USB Dongle  
(BSBT4D09BK)



NAP  
role

BT PAN  
Profile connection

XperiaXZ (Android7)



PANU  
roles

<https://kernel.googlesource.com/pub/scm/bluetooth/bluez.git>

bluez/test (python script)  
- *test-nap*

# Learning Linux BT mechanisms (2/2)

<https://kernel.googlesource.com/pub/scm/bluetooth/bluez.git>

bluez/doc/network-api.txt

Network server hierarchy

=====

Service	org.bluez
Interface	org.bluez.NetworkServer1
Object path	/org/bluez/{hci0,hci1,...}
Methods	void Register(string uuid, string bridge)

Register server for the provided UUID. Every new connection to this server will be added the bridge

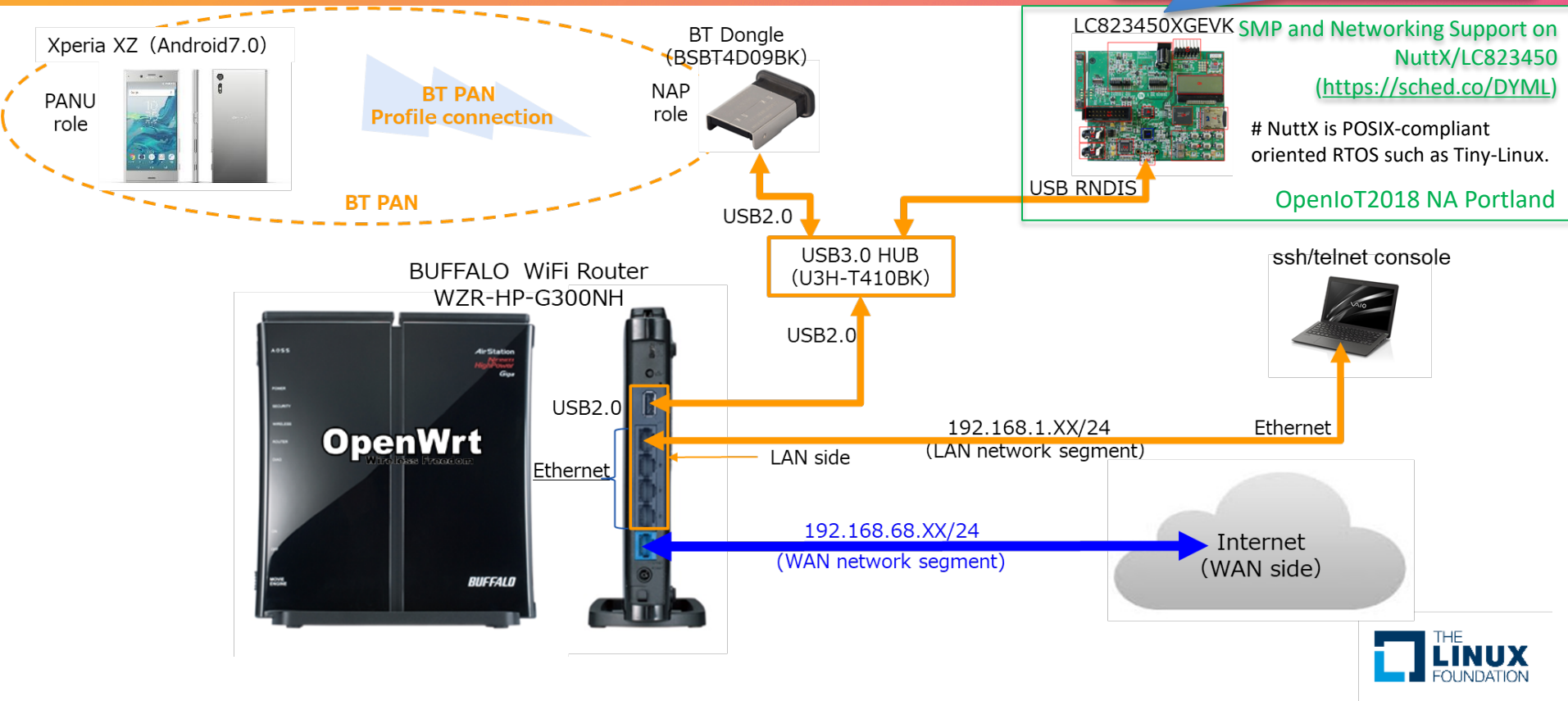
This API is the essence of NAP role for BT PAN connection

'dbus-send' command can invoke this API



# Overview

We have an update of RNDIS in second part





# Hardware Requirement

- OpenWrt supporting hardware with USB port
  - BUFFALO WZR-HP-G300NH (USB 2.0 port)

You can see <https://openwrt.org/toh/start>



The screenshot shows the OpenWrt website's 'Table of Hardware' page. The page lists various hardware devices supported by OpenWrt, including routers, switches, and other network equipment. The table includes columns for the device name, model, processor, memory, and other specifications. The table is organized into a grid with multiple rows and columns, providing a comprehensive overview of the supported hardware.

Device	Model	Processor	Memory	Other Features
Asus	WRT-1900P	Intel Atom	512MB	USB 2.0, 4x LAN, 1x WAN
Buffalo	WZR-HP-G300NH	Realtek	64MB	USB 2.0, 4x LAN, 1x WAN
Linksys	E2500	Realtek	64MB	USB 2.0, 4x LAN, 1x WAN
Mikrotik	RB2011	ARM	512MB	USB 2.0, 4x LAN, 1x WAN
Netgear	R7800	Intel Atom	512MB	USB 2.0, 4x LAN, 1x WAN
TP-Link	TL-WR7400	Realtek	64MB	USB 2.0, 4x LAN, 1x WAN
Ubiquiti	ER-X	ARM	512MB	USB 2.0, 4x LAN, 1x WAN
ZyXEL	US68	Realtek	64MB	USB 2.0, 4x LAN, 1x WAN

- BT USB Dongle supporting PAN profile
  - BUFFALO BT USB Dongle BSBT4D09BK

# Software Requirement

- OpenWrt Firmware for WZR-HP-G300NH with USB RNDIS configuration

I used the below commit at that time .

```
commit 8722c52b41d551e768b3cc46049afb6657099d59
Author: Christoph Krapp <achterin@googlemail.com>
Date: Tue Jul 3 11:06:20 2018 +0200

ath79: remove bs-partition ro-flag for UniFi AC devices

This removes the read-only flag from the bs (bootselect) partition
on UniFi AC devices. This allows to correct the indicator from which
partition the device is booting its kernel from.

See also:
- freifunk-gluon/gluon#1301
- https://bugs.lede-project.org/index.php?do=details&task\_id=662

Signed-off-by: Christoph Krapp <achterin@googlemail.com>
```

<https://git.opensrt.org/openwrt/openwrt.git/>  
commit 8722c52b41d551e768b3cc46049afb6657099d59  
Author: Christoph Krapp <achterin@googlemail.com>  
Date: Tue Jul 3 11:06:20 2018 +0200

```
.config - OpenWrt Configuration

OpenWrt Configuration
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty
submenus ----). Highlighted letters are hotkeys. Pressing <Y>
includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to
exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]

Target System (Atheros AR7xxx/AR9xxx) --->
Subtarget (Generic) --->
Target Profile (Buffalo WZR-HP-G300NH) --->
Target Images --->
Global build settings --->
[ ] Advanced configuration options (for developers) ----
[ ] Build the OpenWrt Image Builder
[ ] Build the OpenWrt SDK
[ ] Package the OpenWrt-based Toolchain
[*] Image configuration --->

<Select> <Exit> <Help> <Save> <Load>
```

# Create OpenWrt Firmware

```
$ git clone https://git.openwrt.org/openwrt/openwrt.git/
```

```
$ cd openwrt
```

```
$ ./scripts/feeds update -a
```

```
$ ./scripts/feeds install -a
```

```
$ cp enable_btpan_usbndis_for_wzr-hz-g300nh.seed .config
```

```
$ make defconfig
```

```
$ make
```

These two command should execute once so that a needed software package can be selected.

I provide config.seed file at the bottom of this slide

Generated firmware exists on `./bin/targets/ar71xx/generic/` directory

# Write OpenWrt firmware

- Firmware update is done from Web GUI
  - BUFFALO's original firmware Web GUI can accept “**openwrt-ar71xx-generic-wzr-hp-g300nh-squashfs-factory.bin**” image
  - After OpenWrt firmware is run, firmware update is done from the OpenWrt Web GUI (Luci) using “**openwrt-ar71xx-generic-wzr-hp-g300nh-squashfs-sysupgrade.bin**” image

# *bluetoothctl* utility flexibility

Please note that “*bluetoothctl*” is the file name of Bluetooth utility program.

## Interactive mode

```
root@OpenWrt:~# bluetoothctl
Agent registered
[bluetooth]# power on
[CHG] Controller 00:1B:DC:06:61:D4 Class: 0x00020000
Changing power on succeeded
[CHG] Controller 00:1B:DC:06:61:D4 Powered: yes
[bluetooth]# quit
root@OpenWrt:~#
```

“root@OpenWrt:~#” is shell prompt. “[Bluetooth]#” is the prompt of *bluetoothctl*.

## Single command line

```
root@OpenWrt:~# bluetoothctl power on
[CHG] Controller 00:1B:DC:06:61:D4 Class: 0x00020000
Changing power on succeeded
[CHG] Controller 00:1B:DC:06:61:D4 Powered: yes
root@OpenWrt:~#
```

I impressed *bluetoothctl* utility consideration for both use cases.

I applied my sample control application in my company after I knew that.

# How to connect BT PAN by hand (1/6)

- ssh login to OpenWrt and set NAP UUID to *bluetoothd*

Service	org.bluez
Interface	org.bluez.NetworkServer1
Object path	/org/bluez/{hci0,hci1,...}
Methods	void Register(string uuid, string bridge)

Register server for the provided UUID. Every new connection to this server will be added the bridge

```
root@OpenWrt:~# dbus-send --system \  
    --dest=org.bluez /org/bluez/hci0 \  
    --type=method_call \  
    org.bluez.NetworkServer1.Register \  
    string:"00001116-0000-1000-8000-00805f9b34fb" \  
    string:"br-lan"
```

```
root@OpenWrt:~#
```



# Now *bluetoothd* supports NAP role (2/6)

```
[bluetooth]# show
```

```
Controller 00:1B:DC:06:61:D4 (public)
```

```
  Name: BlueZ 5.49
```

```
  Alias: BlueZ 5.49
```

```
  Class: 0x00020000
```

```
  Powered: yes
```

```
  Discoverable: no
```

```
  Pairable: yes
```

```
  UUID: Generic Attribute Profile (00001801-0000-1000-8000-00805f9b34fb)
```

```
  UUID: NAP (00001116-0000-1000-8000-00805f9b34fb)
```

```
  UUID: A/V Remote Control (0000110e-0000-1000-8000-00805f9b34fb)
```

```
  UUID: PnP Information (00001200-0000-1000-8000-00805f9b34fb)
```

```
  UUID: A/V Remote Control Target (0000110c-0000-1000-8000-00805f9b34fb)
```

```
  UUID: Generic Access Profile (00001800-0000-1000-8000-00805f9b34fb)
```

```
  Modalias: usb:v1D6Bp0246d0531
```

```
  Discovering: no
```

```
[bluetooth]#
```

Service Discovery

<https://www.bluetooth.com/specifications/assigned-numbers/service-discovery/>

# Set trusted device in advance (3/6)

```
root@OpenWrt:~# bluetoothctl
```

```
[bluetooth]# trust 00:02:5B:00:A5:A5
```


```
[CHG] Device 00:02:5B:00:A5:A5 Trusted: yes
```

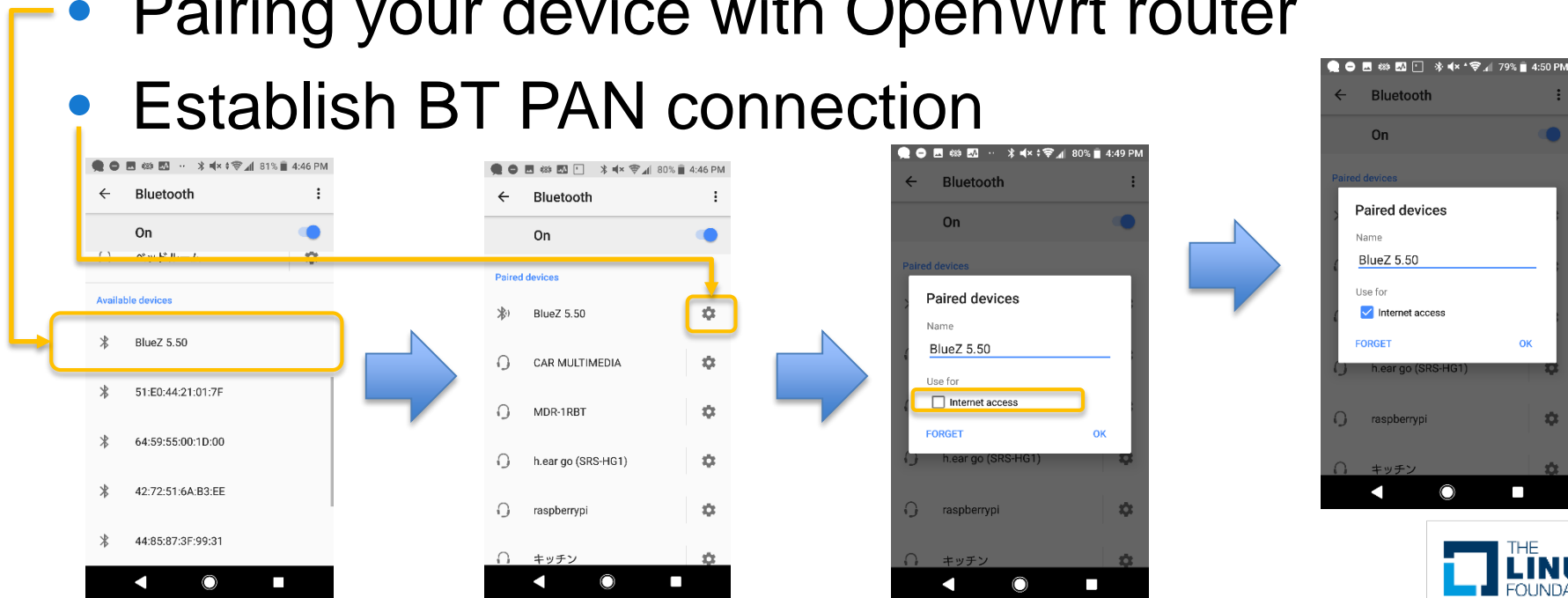
```
Changing 00:02:5B:00:A5:A5 trust succeeded
```

```
[bluetooth]#
```

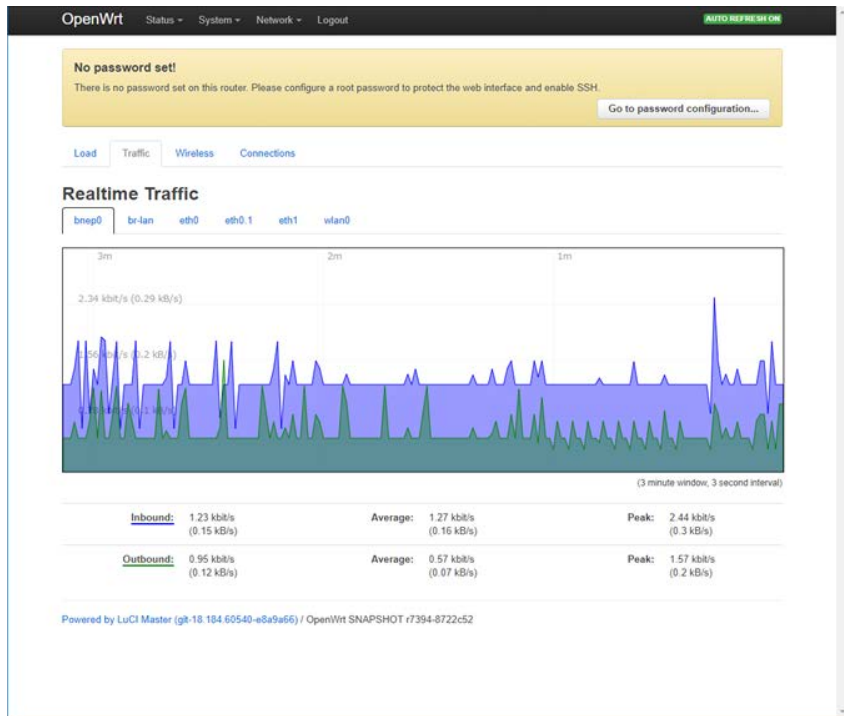
Specify the MAC address of BT PANU  
role device to be connected (e.g: low-  
end device such as second part)

# Connect your device to BT on OpenWrt (4/6)

- Make BT discoverable on OpenWrt router 
- Pairing your device with OpenWrt router
- Establish BT PAN connection



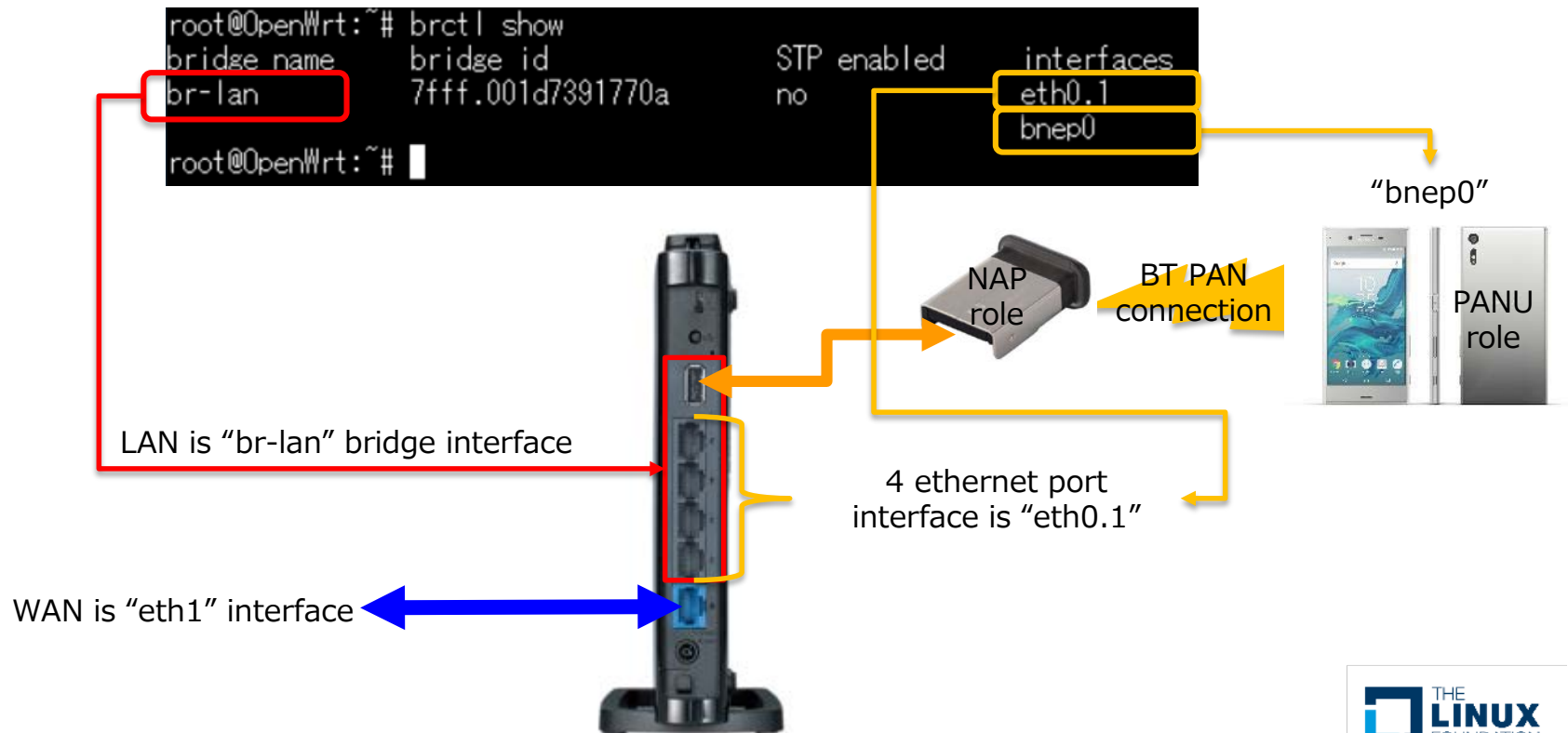
# bnep0 appears on BT PAN router (5/6)



When PANU role device connects to router, bnep0 interface appears on BT PAN router (NAP role) . This node is added “br-lan” bridge interface to which ethernet LAN port connects.

It means BT PANU device and ethernet LAN connected device have same LAN side IP network.

# “bnep0” interface belongs to “br-lan” bridge (6/6)



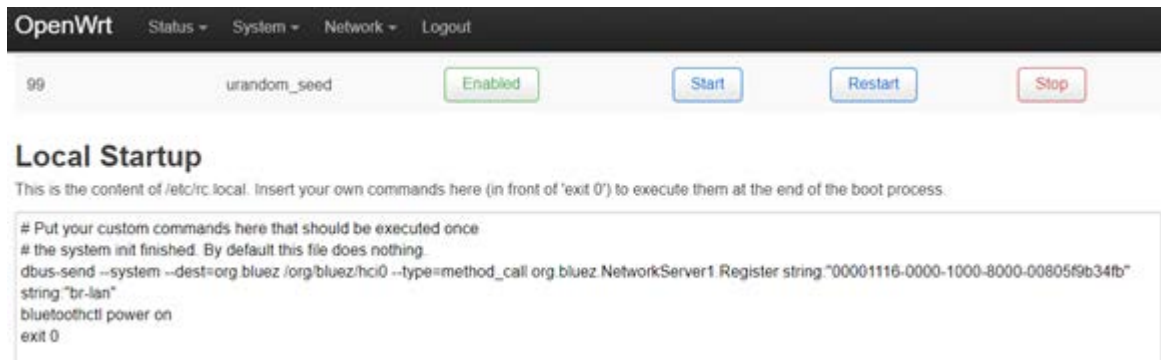
# Persistence of Pairing Information

- Nothing to do for BT pairing
- Once BT pairing is completed by hand, *bluetoothd* (BlueZ daemon) will save its connection data.



# Make NAP work at cold start

- Two things need to be executed at startup time on OpenWrt router to enable BT NAP role
  - Set NAP role to *bluetoothd*
  - BT Dongle Power ON
- “Local Startup” is used for the above operation as shell scripts.
  - “Local Startup” locates Web GUI’s “System” tab -> “Startup” screen on the bottom.



# Introduction to low-end devices running NuttX with OpenWrt router

# Biography (Masayuki Ishikawa)



Senior Software Engineer  
at Sony Home Entertainment and Sound Products Inc.

## Technical background

- 3D graphics, home networking, Internet-to-Home, Embedded Systems

## Product development

- Portable Media Player (Linux/Android)
- Digital voice recorder, music player (NuttX)

## Public talks

- Arm Techcon 2016, ELC2017NA, OpenIoT2018NA, NuttX2019

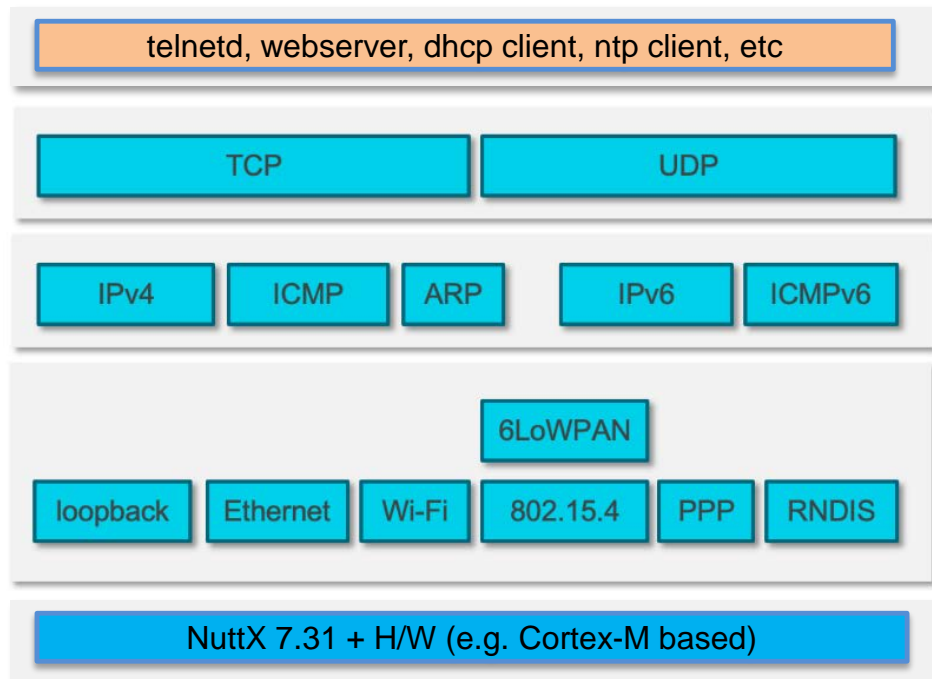
# Introduction to NuttX networking features

## ■ Motivation

- Confirm interoperability between OpenWrt and NuttX by running network applications on resource limited devices

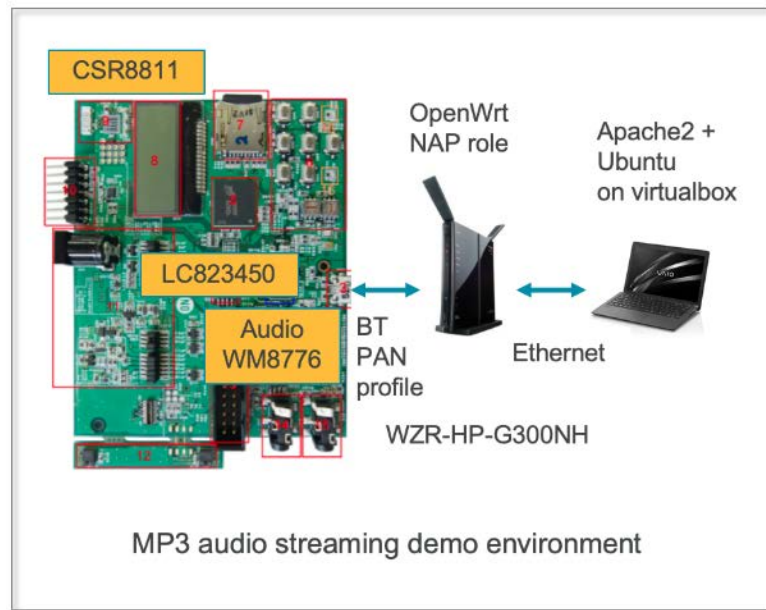
## ■ Features

- Ethernet and IEEE 802.11 Full MAC
- 6LoWPAN for IEEE 802.15.4 MAC
- USB RNDIS, CDC-ECM
- SLIP, TUN/PPP, loopback devices
- IPv4, IPv6, TCP, UDP, ARP, ICMP, ICMPv6, IGMPv2
- BSD compatible socket layer
- DNS name resolution / NetDB
- User socket

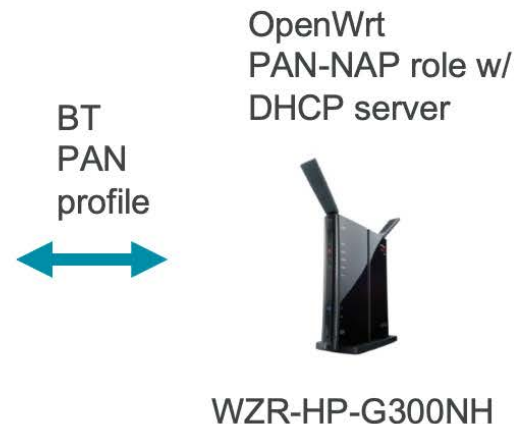
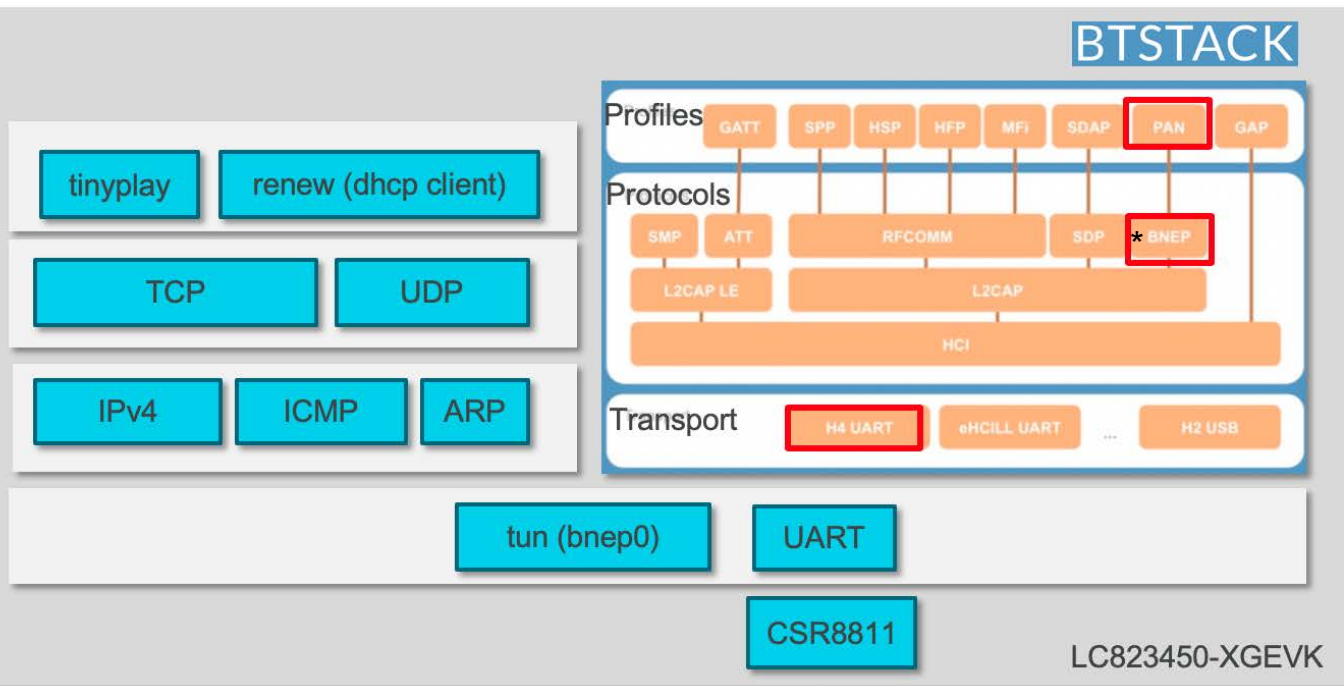


# How to run Bluetooth on NuttX

- Port the BTstack by Bluekitchen to NuttX
  - Based on posix-h4 with H/W flow control
  - UART speed : 921600 baud
  - Free for non-commercial use
- Add TAP mode to the NuttX tun driver
  - TAP mode is used for network bridge
  - NOTE: TUN mode is used for network routing
- HCI\_RESET issue in SMP mode
  - CSR's mode change with HCI\_RESET is tricky
  - Still unstable in SMP mode



# Software stack for Bluetooth networking



\*BNEP: Bluetooth Network Encapsulation Protocol

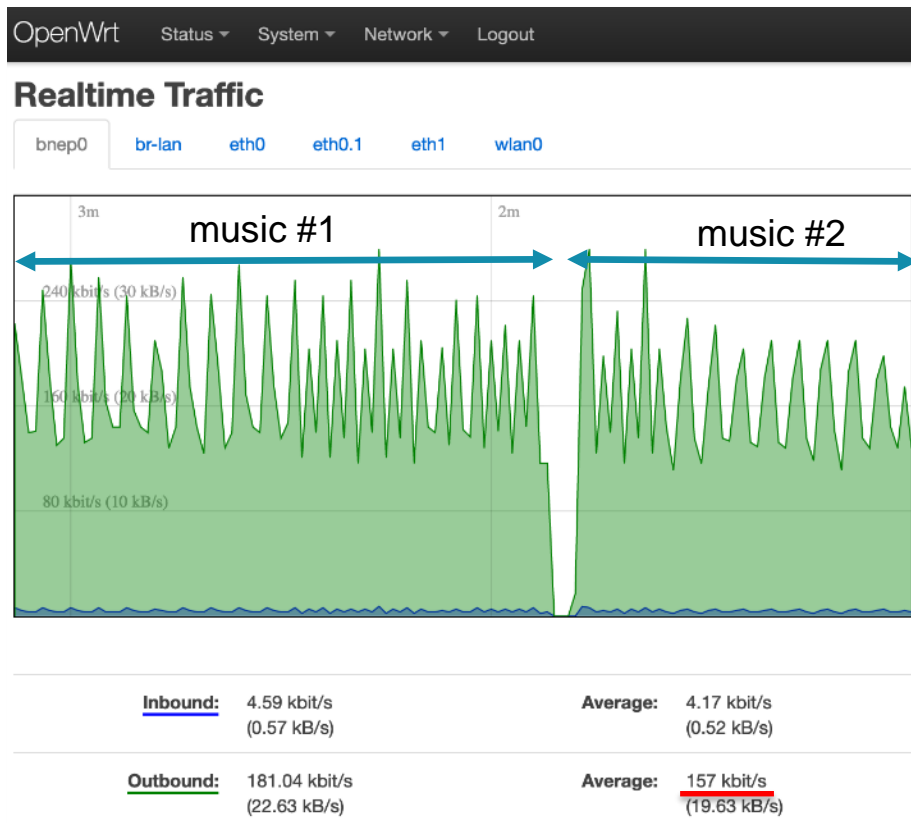


# BTstack log example

```
H4 device: /dev/ttyS1
[2019-06-27 12:12:41.950] LOG -- bnep.c.1582: BNEP_REGISTER_SERVICE mtu 1691
[2019-06-27 12:12:41.950] LOG -- l2cap.c.3387: L2CAP_REGISTER_SERVICE psm 0xf mtu 65535
[2019-06-27 12:12:41.950] LOG -- hci.c.2750: hci_power_control: 1, current mode 0
[2019-06-27 12:12:42.170] LOG -- btstack_uart_block_posix.c.189: h4 set baudrate 115200
[2019-06-27 12:12:42.280] LOG -- hci.c.3797: BTSTACK_EVENT_STATE 1
[2019-06-27 12:12:42.490] LOG -- hci.c.1077: Resend HCI Reset
[2019-06-27 12:12:42.700] LOG -- hci.c.1077: Resend HCI Reset
[2019-06-27 12:12:42.810] LOG -- hci.c.1878: Manufacturer: 0x000a
Local version information:
- HCI Version      0x0006
- HCI Revision     0x2031
```

```
[2019-06-27 12:12:56.990] LOG -- bnep.c.1235: L2CAP_EVENT_CHANNEL_OPENED for BLUETOOTH_PRO
[2019-06-27 12:12:57.000] LOG -- bnep.c.1259: L2CAP_EVENT_CHANNEL_OPENED: outgoing connect
[2019-06-27 12:12:57.010] LOG -- bnep.c.694: bnep_max_frame_size_for_l2cap_mtu: 1691 -> 1
[2019-06-27 12:12:57.070] LOG -- bnep.c.1110: BNEP_CONTROL: Type: 2, size: 3, is extension
[2019-06-27 12:12:57.070] LOG -- bnep.c.879: BNEP_CONNECTION_RESPONSE: Channel established
[2019-06-27 12:12:57.070] LOG -- bnep.c.79: BNEP_EVENT_CHANNEL_OPENED status 0x00 bd_addr:
BNEP connection open succeeded to 00:1B:DC:06:86:59 source UUID 0x1115 dest UUID: 0x1116,
[2019-06-27 12:12:57.070] LOG -- btstack_network.c.264: BNEP device "bnep0" allocated
Network Interface bnep0 activated
```

# MP3 streaming via Bluetooth

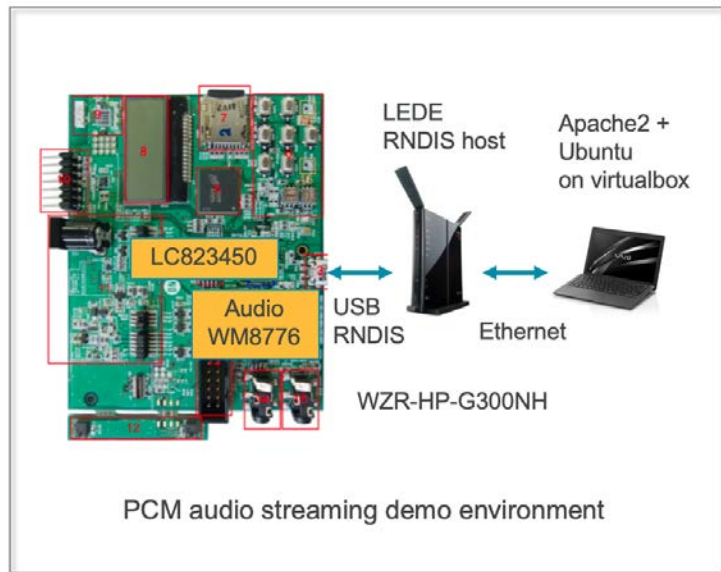


```
56 bytes from 192.168.1.220: icmp_seq=3 time=20 ms
56 bytes from 192.168.1.220: icmp_seq=4 time=20 ms
56 bytes from 192.168.1.220: icmp_seq=5 time=20 ms
56 bytes from 192.168.1.220: icmp_seq=6 time=20 ms
56 bytes from 192.168.1.220: icmp_seq=7 time=20 ms
56 bytes from 192.168.1.220: icmp_seq=8 time=20 ms
56 bytes from 192.168.1.220: icmp_seq=9 time=10 ms
10 packets transmitted, 10 received, 0% packet loss, time 10100
nsh> ifconfig
lo      Link encap:Local Loopback at UP
        inet addr:127.0.0.1 DRaddr:127.0.0.1 Mask:255.0.0.0

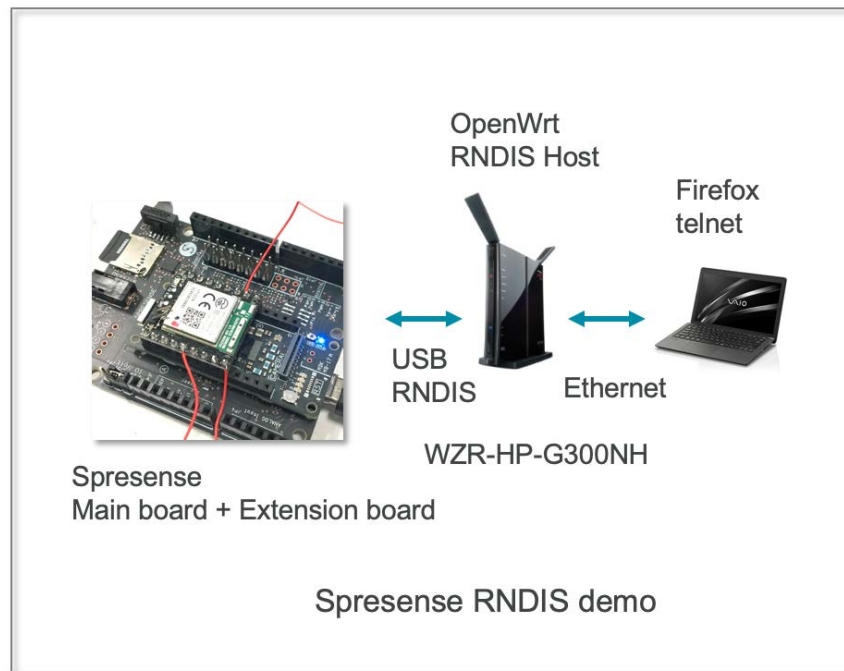
bnep0   Link encap:Ethernet HWaddr 00:02:5b:00:a5:a5 at UP
        inet addr:192.168.1.156 DRaddr:192.168.1.1 Mask:255.255.255.0

        IPv4    TCP    UDP    ICMP
Received      2131  202b  003e  0014
Dropped       00b3  061b  0000  0000
IPv4          VHL:  0002  Frg:  0000
Checksum      0000  0000  0000  ----
TCP           ACK:  0000  SYN:  0000
              RST:  0002  0002
Type          0000  ----  ----  0000
Sent          1a2a  1a14  0002  0014
Rexmit        ----  0005  ----  ----
nsh> tinyplay http://192.168.1.220/~ishikawa/audio/sample2.mp3
tinyplay [14:140]
nsh> fmt=mp3 ch=2 freq=44100
```

# Running RNDIS on NuttX



LC823450XGEVK + LEDE (RNDIS) at OpenIoT 2018

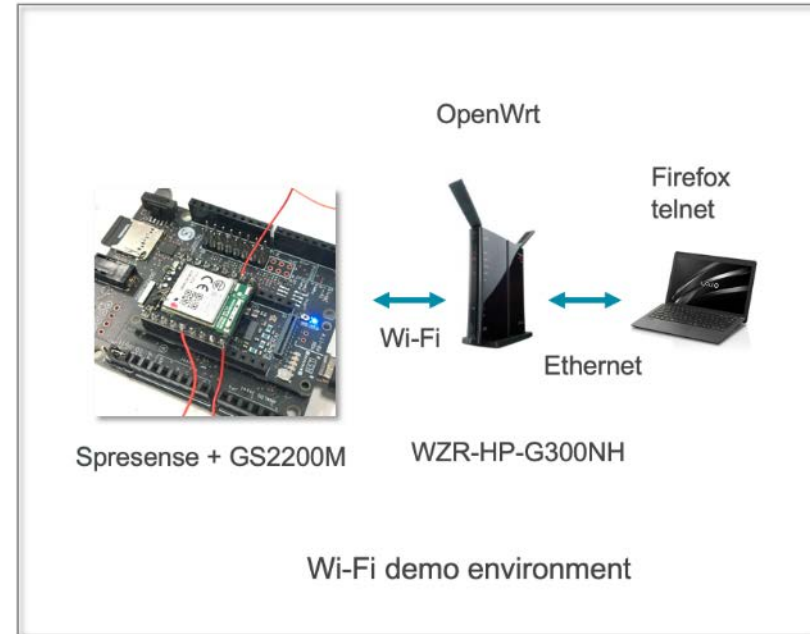


Spresense + OpenWrt (RNDIS) at ELC2019NA



# Working with Wi-Fi on Spresense

- Wi-Fi module : Telit GS2200M
  - Radio protocols: 802.11b/g/n (2.4GHz)
  - Interface : SPI 10MHz with DMA
- Implement GS2200M driver from scratch\*
  - Based on the NuttX usrsock
  - Both STA and AP modes are supported
  - Fix cxd56\_gpoint.c for interrupt handling
  - TCP and UDP are supported
- Modify the uIP webserver app for NuttX
  - Add a directory listing feature



\* The code is available at <https://bitbucket.org/nuttx/nuttx>

# Use case for Webserver via Wi-Fi

Network applications

DHCP client

DNS client

telnetd

webserver

gs220m daemon

/dev/mmcSD0

/dev/ursock

/dev/gS2200m

SDHCI

micro SD

Interrupt

SPI

GS2200M


The latest NuttX upstream + Spresense

OpenWrt

Wi-Fi

WZR-HP-G300NH

Web browsing with Firefox



Name	Last modified	Size
<a href="#">..</a>	-	-
<a href="#">audio/</a>	2019-07-05 16:05	-
<a href="#">ps.txt</a>	2019-07-05 17:06	814
<a href="#">.fsevents/</a>	2019-07-05 17:48	-
<a href="#">.Spotlight-V100/</a>	2019-07-03 09:24	-
<a href="#">spresense_header_code.jpg</a>	2019-07-05 17:48	107550
<a href="#">.Trashes/</a>	2019-07-05 16:04	-
<a href="#">hello.txt</a>	2019-07-05 16:14	8

uIP web server (NuttX 7.30 d228dd3 Jul 5 2019 17:49:54)

VirtualBox + Ubuntu

# Conclusion

- This work shows how to easily extend an OpenWrt router
- Also, NuttX networking is feasible on resource limited devices



# Reference (1/3)

- I knew about LEDE at ELC2017 session in Portland

OpenWrt/LEDE: when two become one, presented by Florian Fainelli, (<https://sched.co/9luP>)

- [https://elinux.org/images/0/0a/ELC\\_OpenWrt\\_LEDE.pdf](https://elinux.org/images/0/0a/ELC_OpenWrt_LEDE.pdf)
- <http://events17.linuxfoundation.org/events/embedded-linux-conference/program/slides>

# Reference (2/3)



Developing Audio Products with Cortex-M3/NuttX/C++11  
(<https://sched.co/900s>)

(ELC2017 North America)



SMP and Networking Support on NuttX/LC823450  
(<https://sched.co/DYML>)

(OpenIoT2018 North America)

# Reference (3/3)

## OpenWrt documents

■ Table of Hardware (supported hardware list)

<https://openwrt.org/toh/start>

■ Official Documents starting point

<https://openwrt.org/docs/start>

■ Developer Guide

<https://openwrt.org/docs/guide-developer/start>

■ Creating packages

<https://openwrt.org/docs/guide-developer/packages>

OpenWrt manages software components on a package basis. For this reason, it is better to create a package to import (Porting software component to OpenWrt) . This URL explains about it. The template of Makefile realizes easy porting explanation by the example of bridge package.

■ Quick Image Building Guide (Image build guide)

<https://openwrt.org/docs/guide-developer/quickstart-build-images>

■ BUFFALO product information

Wi-Fi router

<http://buffalo.jp/products/catalog/network/wzr-hp-g300nh/>

BT Dongle

<http://buffalo.jp/product/peripheral/wireless-adapter/bsbt4d09bk/>

■ IBM WATSON IOT iot-raspberrypi

<https://github.com/ibm-watson-iot/device-raspberrypi>

■ dbus

<https://dbus.freedesktop.org/doc/api/html/index.html>

<https://www.freedesktop.org/wiki/Software/dbus/>

■ dbus-monitor

<https://dbus.freedesktop.org/doc/dbus-monitor.1.html>

dump D-Bus message



**Any Questions?**



**Thank you for your  
participation and  
interest**

# Supplemental Material

- What's OpenWrt
- Next two slides come from our OpenIoT2018 North America “SMP and Networking support on NuttX / LC823450”.



# Introduction to LEDE

SONY

- Motivation

- Build a shareable network testing environment for NuttX

- Software

- LEDE project as of ELC2017 session
- The project was forked from OpenWRT that is famous OSS for the router world as a turn key solution but they became one again (at the beginning of 2018)

- Hardware

- WZR-HP-G300NH (buffalo) Wi-Fi router with USB 2.0 port

WZR-HP-G300NH



# Announcing the OpenWrt/LEDE merge

Installing and Using LEDE created Jan 3 last reply Jan 13 45 replies 18.6k views 17 users 82 likes 16 links

jow SysAdmin Jan 3

Both the OpenWrt and LEDE projects are happy to announce their unification under the OpenWrt name. After long and sometimes slowly moving discussions about the specifics of the re-merge, with multiple similar proposals but little subsequent action, we're happy to announce that both projects are about to execute the final steps of the merger.



The new, unified OpenWrt project will be governed under the rules established by the LEDE project. Active members of both the former LEDE and OpenWrt projects will continue working on the unified OpenWrt.

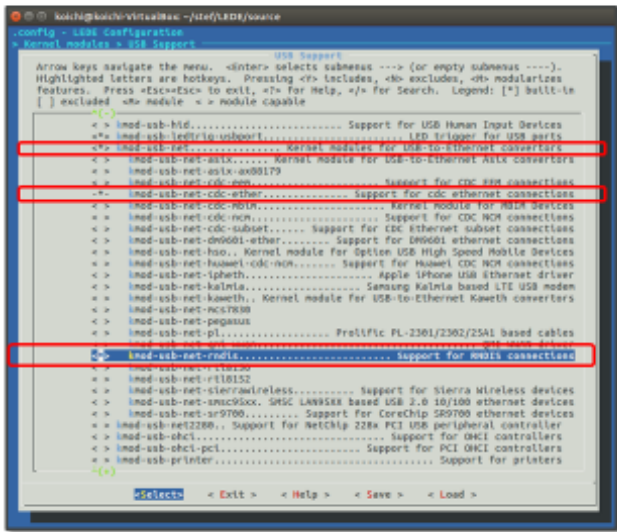
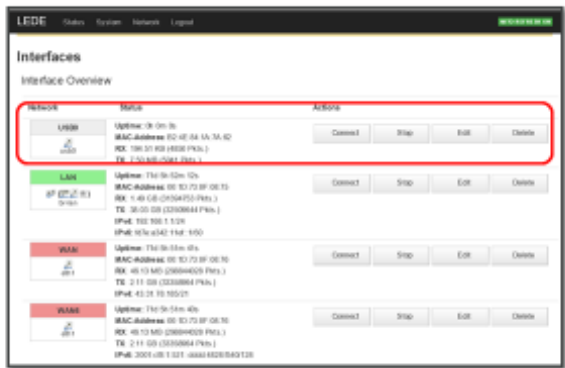
LEDE's fork and subsequent re-merge into OpenWrt will not alter the overall technical direction taken by the unified project. We will continue to work on improving stability and release maintenance while aiming for frequent minor releases to address critical bugs and security issues like we did with LEDE 17.01 and its four point releases until now.

Old pre-15.05 OpenWrt CC releases will not be supported by the merged project anymore, leaving these releases without any future security or bug fixes. The OpenWrt CC 15.05 release series will receive a limited amount of security and bug fixes, but is not yet fully integrated in our release automation, so binary releases are lacking behind for now.

The LEDE 17.01 release will continue to get full security and bug fix support for both source code and binary releases. We are planning a new major release under the new name in the next few months.

## Support RNDIS on LEDE

- How to setup
  - Modify configuration
  - Add network USB0 (RNDIS) via LuCI
  - Change the network setting of USB0





# Embedded Linux Conference

North America

# enable\_btpan\_usbrndis\_for\_wzr-hz-g300nh.seed (1/2)

```
CONFIG_TARGET_ar71xx=y
CONFIG_TARGET_ar71xx_generic=y
CONFIG_TARGET_ar71xx_generic_DEVICE_WZRHPG300NH=y
CONFIG_PACKAGE_bluez-daemon=y
CONFIG_PACKAGE_bluez-libs=y
CONFIG_PACKAGE_bluez-utils=y
CONFIG_PACKAGE_bluez-utils-extra=y
CONFIG_PACKAGE_dbus=y
CONFIG_PACKAGE_dbus-utils=y
CONFIG_PACKAGE_glib2=y
CONFIG_PACKAGE_kmod-bluetooth=y
CONFIG_PACKAGE_kmod-crypto-aead=y
CONFIG_PACKAGE_kmod-crypto-cmac=y
CONFIG_PACKAGE_kmod-crypto-ecb=y
CONFIG_PACKAGE_kmod-crypto-ecdh=y
CONFIG_PACKAGE_kmod-crypto-hash=y
CONFIG_PACKAGE_kmod-crypto-kpp=y
CONFIG_PACKAGE_kmod-crypto-manager=y
CONFIG_PACKAGE_kmod-crypto-null=y
CONFIG_PACKAGE_kmod-crypto-pcompress=y
CONFIG_PACKAGE_kmod-hid=y
CONFIG_PACKAGE_kmod-input-core=y
CONFIG_PACKAGE_kmod-input-evdev=y
CONFIG_PACKAGE_kmod-lib-crc16=y
CONFIG_PACKAGE_kmod-mii=y
CONFIG_PACKAGE_kmod-regmap-core=y
CONFIG_PACKAGE_kmod-usb-net=y
CONFIG_PACKAGE_kmod-usb-net-cdc-ether=y
CONFIG_PACKAGE_kmod-usb-net-rndis=y
CONFIG_PACKAGE_libattr=y
CONFIG_PACKAGE_libdbus=y
CONFIG_PACKAGE_libexpat=y
CONFIG_PACKAGE_libffi=y
CONFIG_PACKAGE_libical=y
CONFIG_PACKAGE_libiinfo-lua=y
```

```
CONFIG_PACKAGE_liblua=y
CONFIG_PACKAGE_liblucihttp=y
CONFIG_PACKAGE_liblucihttp-lua=y
CONFIG_PACKAGE_libncurses=y
CONFIG_PACKAGE_libreadline=y
CONFIG_PACKAGE_librt=y
CONFIG_PACKAGE_libubus-lua=y
CONFIG_PACKAGE_lua=y
CONFIG_PACKAGE_luci=y
CONFIG_PACKAGE_luci-app-firewall=y
CONFIG_PACKAGE_luci-app-opkg=y
CONFIG_PACKAGE_luci-base=y
CONFIG_PACKAGE_luci-lib-ip=y
CONFIG_PACKAGE_luci-lib-jsonc=y
CONFIG_PACKAGE_luci-lib-nixio=y
CONFIG_PACKAGE_luci-mod-admin-full=y
CONFIG_PACKAGE_luci-mod-network=y
CONFIG_PACKAGE_luci-mod-status=y
CONFIG_PACKAGE_luci-mod-system=y
CONFIG_PACKAGE_luci-proto-ipv6=y
CONFIG_PACKAGE_luci-proto-ppp=y
CONFIG_PACKAGE_luci-theme-bootstrap=y
CONFIG_PACKAGE_rpcd=y
CONFIG_PACKAGE_rpcd-mod-rrdns=y
CONFIG_PACKAGE_terminfo=y
CONFIG_PACKAGE_uhttpd=y
CONFIG_PACKAGE_zlib=y
# CONFIG_TARGET_ath79_generic is not set
# CONFIG_TARGET_ath79_generic_DEVICE_8dev_carambola2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_adtran_bsap1800-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_adtran_bsap1840 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_alfa-network_ap121f is not set
# CONFIG_TARGET_ath79_generic_DEVICE_aruba_ap-105 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_avm_fritz300e is not set
# CONFIG_TARGET_ath79_generic_DEVICE_avm_fritz4020 is not set
```

```
# CONFIG_TARGET_ath79_generic_DEVICE_buffalo_bhr-4grv is not set
# CONFIG_TARGET_ath79_generic_DEVICE_buffalo_bhr-4grv2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_buffalo_wzr-hp-ag300h is not set
# CONFIG_TARGET_ath79_generic_DEVICE_buffalo_wzr-hp-g302h-a1a0 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_buffalo_wzr-hp-g450h is not set
# CONFIG_TARGET_ath79_generic_DEVICE_comfast_cf-e110n-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_comfast_cf-e120a-v3 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_comfast_cf-e314n-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_comfast_cf-e5 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_comfast_cf-wr650ac-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_comfast_cf-wr650ac-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_devolo_dvl1200e is not set
# CONFIG_TARGET_ath79_generic_DEVICE_devolo_dvl1200i is not set
# CONFIG_TARGET_ath79_generic_DEVICE_devolo_dvl1750c is not set
# CONFIG_TARGET_ath79_generic_DEVICE_devolo_dvl1750e is not set
# CONFIG_TARGET_ath79_generic_DEVICE_devolo_dvl1750i is not set
# CONFIG_TARGET_ath79_generic_DEVICE_devolo_dvl1750x is not set
# CONFIG_TARGET_ath79_generic_DEVICE_dlink_dir-825-b1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_dlink_dir-825-c1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_dlink_dir-835-a1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_dlink_dir-842-c1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_dlink_dir-842-c2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_dlink_dir-842-c3 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_dlink_dir-859-a1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_elecom_wrc-1750ghbk2-i is not set
# CONFIG_TARGET_ath79_generic_DEVICE_elecom_wrc-300ghbk2-i is not set
# CONFIG_TARGET_ath79_generic_DEVICE_embeddedwireless_dorin is not set
# CONFIG_TARGET_ath79_generic_DEVICE_engenius_ecb1750 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_engenius_epg5000 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_engenius_ews511ap is not set
# CONFIG_TARGET_ath79_generic_DEVICE_etactica_eg200 is not set
```



# enable\_btpan\_usbndis\_for\_wzr-hz-g300nh.seed (2/2)

```
# CONFIG_TARGET_ath79_generic_DEVICE_glinet_gl-ar150 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_glinet_gl-ar300m-lite is not set
# CONFIG_TARGET_ath79_generic_DEVICE_glinet_gl-ar300m-nor is not set
# CONFIG_TARGET_ath79_generic_DEVICE_glinet_gl-ar750s is not set
# CONFIG_TARGET_ath79_generic_DEVICE_glinet_gl-x750 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_iodata_etg3-r is not set
# CONFIG_TARGET_ath79_generic_DEVICE_iodata_wn-ac1167dgr is not set
# CONFIG_TARGET_ath79_generic_DEVICE_iodata_wn-ac1600dgr is not set
# CONFIG_TARGET_ath79_generic_DEVICE_iodata_wn-ac1600dgr2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_iodata_wn-ag300dgr is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ijplus_ja76pf2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_librerouter_librerouter-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_nec_wg1200cr is not set
# CONFIG_TARGET_ath79_generic_DEVICE_nec_wg800hp is not set
# CONFIG_TARGET_ath79_generic_DEVICE_netgear_ex6400 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_netgear_ex7300 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_netgear_wndr3700 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_netgear_wndr3700v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_netgear_wndr3800 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ocedo_koala is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ocedo_raccoon is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ocedo_ursus is not set
# CONFIG_TARGET_ath79_generic_DEVICE_openmesh_om5p-ac-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_pcs_cap324 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_pcs_cr3000 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_pcs_cr5000 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_phicomm_k2t is not set
# CONFIG_TARGET_ath79_generic_DEVICE_pisen_wmb001n is not set
# CONFIG_TARGET_ath79_generic_DEVICE_pisen_wmm003n is not set
# CONFIG_TARGET_ath79_generic_DEVICE_qihoo_c301 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_robinson_wr818 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-a7-v5 is not set
```

```
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c2-v3 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c25-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c5-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c58-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c59-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c6-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c60-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c60-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c7-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c7-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c7-v4 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-c7-v5 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_archer-d50-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe210-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe210-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe210-v3 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe220-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe510-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe510-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe510-v3 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_cpe610-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_re350k-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_re355-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_re450-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_re450-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wdr3600-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wdr4300-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wdr4900-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr1043n-v5 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr1043nd-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr1043nd-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr1043nd-v3 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr1043nd-v4 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr2543-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr710n-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr810n-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr810n-v2 is not set
```

```
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr842n-v1 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr842n-v2 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_tplink_tl-wr842n-v3 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_trendnet_tew-823dru is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_acb-isp is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_airstar is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_bullet-m is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_bullet-m-xw is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_lap-120 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_nanobeam-ac is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_nanostation-ac is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_nanostation-ac-loco is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_nanostation-m is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_nanostation-m-xw is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_rocket-m is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_routerstation is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_routerstation-pro is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_unifi is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_unifiac-lite is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_unifiac-mesh is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_unifiac-mesh-pro is not set
# CONFIG_TARGET_ath79_generic_DEVICE_ubnt_unifiac-pro is not set
# CONFIG_TARGET_ath79_generic_DEVICE_wd_mynet-n750 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_wd_mynet-wifi-rangextender is not set
# CONFIG_TARGET_ath79_generic_DEVICE_winchannel_wb2000 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_xiaomi_mi-router-4q is not set
# CONFIG_TARGET_ath79_generic_DEVICE_yuncore_a770 is not set
# CONFIG_TARGET_ath79_generic_DEVICE_zbtlink_zbt-wd323 is not set
# CONFIG_TARGET_ath79_generic_Default is not set
# CONFIG_TARGET_ath79_nand is not set
# CONFIG_TARGET_ath79_tiny is not set
```